

**TMS320F28030, TMS320F28031,
TMS320F28032, TMS320F28033,
TMS320F28034, TMS320F28035 Piccolo MCU**

Silicon Errata



Literature Number: SPRZ295J
April 2009–Revised February 2012

1	Introduction	4
2	Device and Development Support Tool Nomenclature	4
3	Device Markings	5
4	Known Design Marginality/Exceptions to Functional Specifications	6
5	Documentation Support	23
6	I-to-J Revision History	24
7	H-to-I Revision History	25

List of Figures

1	Example of Device Markings	5
2	Device Nomenclature.....	5

List of Tables

1	Determining Silicon Revision From Lot Trace Code (2803x Devices)	5
2	Advisory List	6

TMS320F2803x Piccolo MCU Silicon Errata

1 Introduction

This document describes the silicon updates to the functional specifications for the TMS320F2803x microcontrollers (MCUs).

The updates are applicable to:

- 56-pin Plastic Quad Flatpack, RSH Suffix
- 64-pin Thin Quad Flatpack, PAG Suffix
- 80-pin Low-Profile Quad Flatpack, PN Suffix

2 Device and Development Support Tool Nomenclature

To designate the stages in the product development cycle, TI assigns prefixes to the part numbers of all [TMS320] DSP devices and support tools. Each TMS320™ DSP commercial family member has one of three prefixes: TMX, TMP, or TMS (e.g., **TMS320F28035**). Texas Instruments recommends two of three possible prefix designators for its support tools: TMDX and TMDS. These prefixes represent evolutionary stages of product development from engineering prototypes (TMX/TMDX) through fully qualified production devices/tools (TMS/TMDS).

TMX	Experimental device that is not necessarily representative of the final device's electrical specifications
TMP	Final silicon die that conforms to the device's electrical specifications but has not completed quality and reliability verification
TMS	Fully qualified production device

Support tool development evolutionary flow:

TMDX	Development-support product that has not yet completed Texas Instruments internal qualification testing
TMDS	Fully qualified development-support product

TMX and TMP devices and TMDX development-support tools are shipped against the following disclaimer:

"Developmental product is intended for internal evaluation purposes."

TMS devices and TMDS development-support tools have been characterized fully, and the quality and reliability of the device have been demonstrated fully. TI's standard warranty applies.

Predictions show that prototype devices (TMX or TMP) have a greater failure rate than the standard production devices. Texas Instruments recommends that these devices not be used in any production system because their expected end-use failure rate still is undefined. Only qualified production devices are to be used.

TI device nomenclature also includes a suffix with the device family name. This suffix indicates the package type (for example, PN) and temperature range (for example, T).

3 Device Markings

Figure 1 provides an example of the 2803x device markings and defines each of the markings. The device revision can be determined by the symbols marked on the top of the package as shown in Figure 1. Some prototype devices may have markings different from those illustrated. Figure 2 shows the device nomenclature.

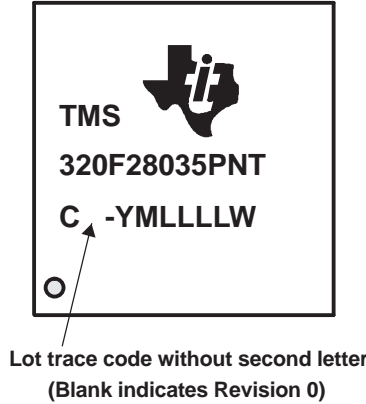


Figure 1. Example of Device Markings

Table 1. Determining Silicon Revision From Lot Trace Code (2803x Devices)

SECOND LETTER IN PREFIX OF LOT TRACE CODE	SILICON REVISION	REVISION ID (0x0883)	COMMENTS
Blank (no second letter in prefix)	Indicates Revision 0	0x0000	This silicon revision is available as TMX.
A	Indicates Revision A	0x0001	This silicon revision is available as TMS.

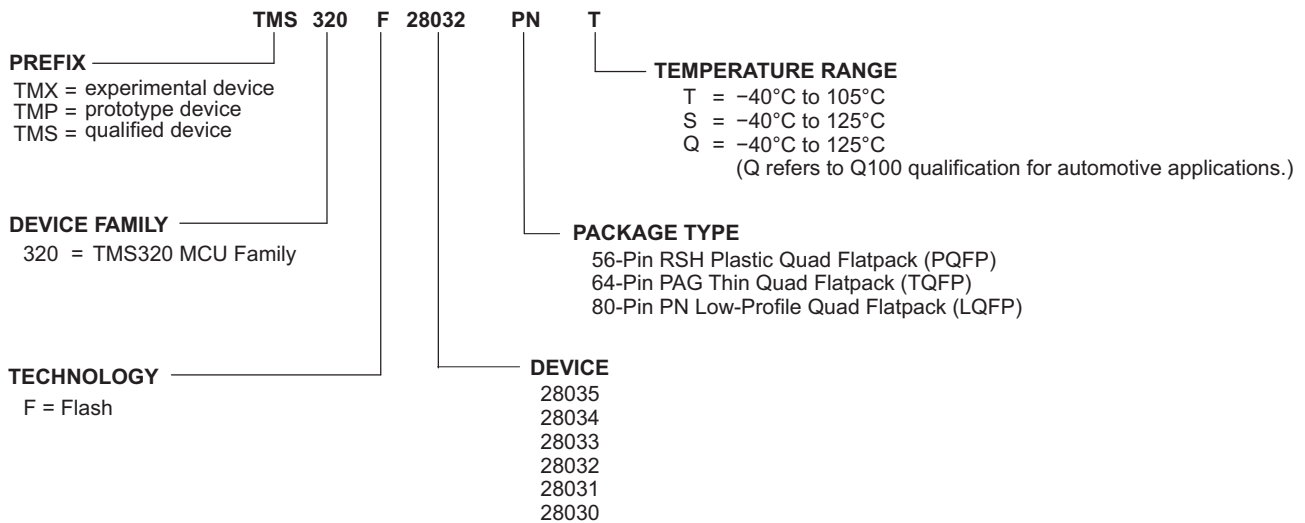


Figure 2. Device Nomenclature

4 Known Design Marginality/Exceptions to Functional Specifications

Table 2. Advisory List

Title	Page
Advisory —ADC: Initial Conversion.....	7
Advisory —ADC: Temperature Sensor Minimum Sample Window Requirement	7
Advisory —ADC: DC Specifications: Linearity Limitation	8
Advisory —ADC: New ADC Control Bits Added to Revision A Silicon.....	8
Advisory —ADC: Out-of-Specification Current Consumption on Power Up.....	9
Advisory —ADC: ADC Result Conversion When Sampling Ends on 14th Cycle of Previous Conversion, ACQPS = 6 or 7	9
Advisory —ADC: Offset Self-Recalibration Requirement	10
Advisory —Memory: Prefetching Beyond Valid Memory	11
Advisory —eCAN: Abort Acknowledge Bit Not Set	12
Advisory —GPIO: GPIO Qualification.....	13
Advisory —Code Security Module: Memory Access Conflicts With Code Security Module (CSM) PWL Registers	14
Advisory —LIN: SCI Operation in Multi-Buffer Mode	15
Advisory —LIN: Possible Data Corruption in Extended Frame Mode	15
Advisory —LIN: LIN Transmission is Affected if Previous Transmission was not Complete due to Bit Error (BE) During Checksum Field in Single Buffer Mode.....	15
Advisory —LIN: TOA3WUP Gets Flagged When New Header is Received During Wait Time of 1.5 s	16
Advisory —LIN: Frame Error (FE) During Checksum Byte Restarts Reception. A Frame Error is Flagged if a New Header is Received.	16
Advisory —LIN: When LIN Slave Adapts Itself to a Faster Incoming Header, No Response Error (NRE)/Time-out Might not get Flagged Within Expected Time Frame.....	16
Advisory —LIN: LIN Rejects a Valid Byte Followed Upon After a False Start Bit.....	17
Advisory —LIN: Incomplete Synch Field Reception Results in Loss of Next Header	17
Advisory —LIN: When a Slave Receives a Wake-up Request With Length > 500 μ s, it Does Not Transmit a Response to the First Header Following the Wake-up.....	17
Advisory —LIN: Incomplete Frame Header Consisting of Only Sync Break Will Cause the Following Complete Header to be not Received by Slave	18
Advisory —Boot-ROM: eCAN Boot Mode Not Available on TMX Samples	19
Advisory —Boot ROM: Flash API [Flash_Erase() Function]	19
Advisory —Zero-Pin Oscillator: Modification to Oscillator Frequency Parameter	20
Advisory —eQEP: Missed First Index Event.....	21
Advisory —eQEP: eQEP Inputs in GPIO Asynchronous Mode	21
Advisory —Watchdog: Incorrect Operation of CPU Watchdog When WDCLK Source is OSCCLKSRC2	22
Advisory —Oscillator: CPU Clock Switching to INTOSC2 May Result in Missing Clock Condition After Reset	22

Advisory	ADC: Initial Conversion
Revision(s) Affected	0
Details	When the ADC conversions are initiated by any source of trigger in either sequential or simultaneous sampling mode, the first sample may not be the correct conversion result.
Workaround(s)	<p>For sequential mode, discard the first sample at the beginning of every series of conversions. For instance, if the application calls for a given series of conversions, SOC0→SOC1→SOC2, to initiate periodically, then set up the series instead as SOC0→SOC1→SOC2→SOC3 and only use the last three conversions, ADCRESULT1, ADCRESULT2, ADCRESULT3, thereby discarding ADCRESULT0.</p> <p>For simultaneous sample mode, discard the first sample of both the A and B channels at the beginning of every series of conversions.</p> <p>User application should validate if this workaround is acceptable in their application.</p> <p>The following is applicable to the revision A silicon:</p> <ul style="list-style-type: none"> • For 30-MHz operation and below, this is fixed completely by writing a 1 to the ADCNONOVERLAP and CLKDIV2EN bits in the ADCTRL2 register. This will give a 30-MHz ADC clock when the CPU clock = 60 MHz, and will only allow the sampling of ADC channels when the ADC is finished with any pending conversion. • For 60-MHz or 40-MHz operation, the first sample deviation is still under characterization by TI. The current recommendation is to observe the rev 0 errata until this characterization is complete.

Advisory	ADC: Temperature Sensor Minimum Sample Window Requirement
Revision(s) Affected	0, A
Details	If the minimum sample window is used (6 ADC clocks at 60 MHz, 116.67 ns), the result of a temperature sensor conversion can have a large error, making it unreliable for the system.
Workaround(s)	<ol style="list-style-type: none"> 1. If double-sampling of the temperature sensor is used to avoid the corrupted first sample issue, the temperature sensor result is valid. This is equivalent to giving the S/H circuit adequate time to charge. 2. In all other conditions, the sample-and-hold window used to sample the temperature sensor should not be less than 550 ns.

Advisory	<i>ADC: DC Specifications: Linearity Limitation</i>
Revision(s) Affected	0, A
Details	The linearity degrades with increasing temperature in the upper half of the transfer function.
Workaround(s)	<p>The impact from this limitation has been addressed in the revision A silicon. The following features have been added:</p> <ol style="list-style-type: none"> 1. ADC clock divider-by-2 enable bit. At 60 MHz, the effective sample rate will be 2.3 MSPS. This offers a 30-MHz ADC and a 60-MHz system clock, and improves linearity. 2. Existing pipeline mode with 4.6 MSPS at 60-MHz system clock will have improved linearity compared to revision 0 silicon. <hr/> <p>NOTE: For 60-MHz operation, there are periodic missing codes, and INL will be bounded by ± 28 LSBs MAX/MIN.</p> <p>For 30-MHz operation, see the <i>TMS320F28030</i>, <i>TMS320F28031</i>, <i>TMS320F28032</i>, <i>TMS320F28033</i>, <i>TMS320F28034</i>, <i>TMS320F28035</i> <i>Piccolo Microcontrollers Data Manual</i> (literature number SPRS584).</p> <hr/>

Advisory	<i>ADC: New ADC Control Bits Added to Revision A Silicon</i>
Revision(s) Affected	A
Details	<p>The following bits/features have been added to the ADC control registers in the revision A silicon [see the <i>TMS320x2802x</i>, <i>2803x Piccolo Analog-to-Digital Converter (ADC) and Comparator Reference Guide</i> (literature number SPRUG5) for more information]:</p> <ol style="list-style-type: none"> 1. Register ADCTL2 at address 0x7101, containing the following bits: <ul style="list-style-type: none"> • CLKDIV2EN, a /2 divide enable that scales CPU Clock by 1/2 for use by the ADC • ADCNONOVERLAP, a bit enable that removes the conversion/sample overlap timing 2. Addition of bit field ONESHOT in the existing ADCSOCPRIORITYCTL register. When enabled, the ONESHOT bit field will only let the first SOC complete if multiple SOCs are received at the same time. This is applicable for all SOCs that are in Round-Robin Priority. SOCs in High-Priority mode are not effected by this function.
Workaround(s)	Not applicable

Advisory *ADC: Out-of-Specification Current Consumption on Power Up*

Revision(s) Affected 0

Details When the ADC module is powered up for the first time, it can consume 2x the current listed (I_{DDA}) in the data manual until a sample is initiated.

Workaround(s) Before the ADC is initially powered up, initiate a single conversion. Once the timing of the conversion has been satisfied, an ADC reset is recommended to restore the native state of the ADC before setting up the ADC as done before this addition. (Code is shown below.) If this amount of current can be tolerated by the system, then no change needs to be made to the setup code. Note that the current will go back to within data manual limits once the first conversion is processed by the ADC.

```

EALLOW;
SysCtrlRegs.PCLKCR0.bit.ADCENCLK=1;           //Enable Clocks to ADC module
AdcRegs.ADCSOCFRCL.bit.SOC0 = 1;              //Issue dummy conversion
asm(" rpt #19 || nop");                         //Wait for conversion to propagate
AdcRegs.SOCPRICTL.bit.SOCPRIORITY = 1;         //Change Priority Control to reset
AdcRegs.SOCPRICTL.bit.SOCPRIORITY = 0;         //round robin pointer
EDIS;

```

Advisory *ADC: ADC Result Conversion When Sampling Ends on 14th Cycle of Previous Conversion, ACQPS = 6 or 7*

Revision(s) Affected 0, A

Details The on-chip ADC takes 13 ADC clock cycles to complete a conversion after the sampling phase has ended. The result is then presented to the CPU on the 14th cycle post-sampling and latched on the 15th cycle into the ADC result registers. If the next conversion's sampling phase terminates on this 14th cycle, the results latched by the CPU into the result register are not assured to be valid across all operating conditions.

Workaround(s) Some workarounds are as follows:

- Due to the nature of the sampling and conversion phases of the ADC, there are only two values of ACQPS (which controls the sampling window) that would result in the above condition occurring—ACQPS = 6 or 7. One solution is to avoid using these values in ACQPS.
- When the ADCNONOVERLAP feature (bit 1 in ADCTRL2 register) is used, the above condition will never be met; so the user is free to use any value of ACQPS desired.
- Depending on the frequency of ADC sampling used in the system, the user can determine if their system will hit the above condition if the system requires the use of ACQPS = 6 or 7. For instance, if the converter is continuously converting with ACQPS = 6, the above condition will never be met because the end of the sampling phase will always fall on the 13th cycle of the current conversion in progress.

Advisory **ADC: Offset Self-Recalibration Requirement**

Revision(s) Affected 0, A**Details** The factory offset calibration from Device_cal() may not ensure that the ADC offset remains within specifications under all operating conditions in the customer's system.**Workaround(s)**

- To ensure that the offset remains within the data sheet's "single recalibration" specifications, perform the AdcOffsetSelfCal() function after Device_cal() has completed and the ADC has been configured.
- To ensure that the offset remains within the data sheet's "periodic recalibration" specifications, perform the AdcOffsetSelfCal() function periodically with respect to temperature drift.

For more details on AdcOffsetSelfCal(), refer to the "ADC Zero Offset Calibration" section of the *TMS320x2802x, 2803x Piccolo Analog-to-Digital Converter (ADC) and Comparator Reference Guide* (literature number [SPRUGE5](#)).

Advisory***Memory: Prefetching Beyond Valid Memory***

Revision(s) Affected

0, A

Details

The C28x CPU prefetches instructions beyond those currently active in its pipeline. If the prefetch occurs past the end of valid memory, then the CPU may receive an invalid opcode.

Workaround

The prefetch queue is 8 x16 words in depth. Therefore, code should not come within 8 words of the end of valid memory. This restriction applies to all memory regions and all memory types (flash, OTP, SARAM) on the device. Prefetching across the boundary between two valid memory blocks is all right.

Example 1: M1 ends at address 0x7FF and is not followed by another memory block. Code in M1 should be stored no farther than address 0x7F7. Addresses 0x7F8-0x7FF should not be used for code.

Example 2: M0 ends at address 0x3FF and valid memory (M1) follows it. Code in M0 can be stored up to and including address 0x3FF. Code can also cross into M1 up to and including address 0x7F7.

Advisory	<i>eCAN: Abort Acknowledge Bit Not Set</i>
Revision(s) Affected	0, A
Details	<p>After setting a Transmission Request Reset (TRR) register bit to abort a message, there are some rare instances where the TRRn and TRSn bits will clear without setting the Abort Acknowledge (AAAn) bit. The transmission itself is correctly aborted, but no interrupt is asserted and there is no indication of a pending operation.</p> <p>In order for this rare condition to occur, all of the following conditions must happen:</p> <ol style="list-style-type: none"> 1. The previous message was not successful, either because of lost arbitration or because no node on the bus was able to acknowledge it or because an error frame resulted from the transmission. The previous message need not be from the same mailbox in which a transmit abort is currently being attempted. 2. The TRRn bit of the mailbox should be set in a CPU cycle immediately following the cycle in which the TRSn bit was set. The TRSn bit remaining set due to incompleteness of transmission satisfies this condition as well. i.e. the TRSn bit could have been set in the past, but the transmission remains incomplete. 3. The TRRn bit must be set in the exact SYSCLKOUT cycle where the CAN module is in idle state for one cycle. The CAN module is said to be in idle state when it is not in the process of receiving/transmitting data. <p>If these conditions occur, then the TRRn and TRSn bits for the mailbox will clear t_{clr} SYSCLKOUT cycles after the TRR bit is set where:</p>
	$t_{clr} = [(\text{mailbox_number}) * 2] + 3 \text{ SYSCLKOUT cycles}$
	<p>The TAn and AAAn bits will not be set if this condition occurs. Normally, either the TA or AA bit sets after the TRR bit goes to zero.</p>
Workaround(s)	<p>When this problem occurs, the TRRn and TRSn bits will clear within t_{clr} SYSCLKOUT cycles. To check for this condition, first disable the interrupts. Check the TRRn bit t_{clr} SYSCLKOUT cycles after setting the TRRn bit to make sure it is still set. A set TRRn bit indicates that the problem did not occur.</p> <p>If the TRRn bit is cleared, it could be because of the normal end of a message and the corresponding TAn or AAAn bit is set. Check both the TAn and AAAn bits. If either one of the bits is set, then the problem did not occur. If they are both zero, then the problem did occur. Handle the condition like the interrupt service routine would expect that the AAAn bit does not need clearing now.</p> <p>If the TAn or AAAn bit is set, then the normal interrupt routine will happen when the interrupt is re-enabled.</p>

Advisory
GPIO: GPIO Qualification

Revision(s) Affected

0, A

Details

If a GPIO pin is configured for "n" SYSCLKOUT cycle qualification period (where $1 \leq n \leq 510$) with "m" qualification samples ($m = 3$ or 6), it is possible that an input pulse of $[n * m - (n - 1)]$ width may get qualified (instead of $n * m$). This depends upon the alignment of the asynchronous GPIO input signal with respect to the phase of the internal prescaled clock, and hence, is not deterministic. The probability of this kind of wrong qualification occurring is "1/n".

Worst-case example:

If $n = 510$, $m = 6$, a GPIO input width of $(n * m) = 3060$ SYSCLKOUT cycles is required to pass qualification. However, because of the issue described in this advisory, the minimum GPIO input width which may get qualified is $[n * m - (n - 1)] = 3060 - 509 = 2551$ SYSCLKOUT cycles.

Workaround(s)

None. Ensure a sufficient margin is in the design for input qualification.

Advisory — Code Security Module: Memory Access Conflicts With Code Security Module (CSM) PWL Registers
www.ti.com

Advisory **Code Security Module: Memory Access Conflicts With Code Security Module (CSM) PWL Registers**

Revision(s) Affected 0, A

Details Reads of addresses 0x3D 7FF8–0x3D 7FFF will activate the CSM, locking the device.

Workaround(s) Do not read these addresses. If these addresses are read, causing the CSM to become active, the device can be unlocked either by first pulling reset or by setting the FORCESEC bit, and then performing the Password Match Flow described in the *TMS320x2803x Piccolo System Control and Interrupts Reference Guide* (literature number [SPRUGL8](#)).

NOTE: For TMS320F2803x devices, the PARTID register is located at 0x3D 7E80 (and not at 0x3D 7FFF, as on the TMS320F2802x devices).

Advisory ***LIN: SCI Operation in Multi-Buffer Mode***

Revision(s) Affected 0, A

Details When the LIN node is in SCI multi-buffer mode and if the buffer length is configured as less than 8, then the first set of data (i.e., programmed number of bytes) is transmitted correctly but the transmission of the first byte of subsequent sets of data is corrupted. The TX_EMPTY flag gets set at the start of the STOP bit transmission instead of at the end of the STOP bit.

This issue is applicable only for buffered SCI mode and for buffer length less than 8 bytes.

Workaround(s) When SCI Buffered mode is selected, always configure the buffer length to “8”.

Advisory ***LIN: Possible Data Corruption in Extended Frame Mode***

Revision(s) Affected 0, A

Details In Extended Frame Mode, during Single Buffer Mode transmission, when the TX Buffer (TD0) is loaded with new data during the STOP bit transmission of the previous data byte, transmission of the next data byte will be corrupted (i.e., start bit of the next data byte will be corrupted).

This issue is applicable only when the Extended Frame Mode is enabled and Single Buffer Mode transmission is selected. This problem will occur on both Master and Slave nodes.

Workaround(s) In Extended Frame Mode, during Single Buffer Mode transmission, always wait for the TX_EMPTY flag before loading the TD0 buffer with the new data.

Advisory ***LIN: LIN Transmission is Affected if Previous Transmission was not Complete due to Bit Error (BE) During Checksum Field in Single Buffer Mode***

Revision(s) Affected 0, A

Details In Single Buffer Mode transmission, if Bit Error occurs during the transmission of the checksum byte, then after the reception of next header, there will be an idle period of $10 T_{bit}$ times after the first data byte transmission. This results in the transmission of 1 byte less than the configured response length and also impacts the slave's No Response Error (NRE) count. This issue is applicable only during Single Buffer Mode transmission. This is applicable for both Master and Slave nodes.

Workaround(s) Issue a software reset upon bit error.

Advisory	<i>LIN: TOA3WUP Gets Flagged When New Header is Received During Wait Time of 1.5 s</i>
Revision(s) Affected	0, A
Details	<p>This issue occurs only when the Sleep feature is used in the application. When a LIN node transmits wake-up signals onto the LIN bus, it follows the sequence below:</p> <ul style="list-style-type: none"> • Transmit first wake-up signal → Wait for TOAWUP flag to get set • Transmit second wake-up signal → Wait for TOAWUP flag to get set • Transmit third wake-up signal → Wait for TOAWUP flag to get set • Wait for 1.5 seconds and then set TOA3WUP flag, if no frame headers are received. <p>Due to this erratum, even if a valid header is received during the wait period of 1.5 seconds, the TOA3WUP flag gets set, triggering an interrupt. This is applicable for both Master and Slave nodes.</p>
Workaround(s)	The TOA3WUP interrupt can be ignored by the application, if acceptable.
Advisory	<i>LIN: Frame Error (FE) During Checksum Byte Restarts Reception. A Frame Error is Flagged if a New Header is Received.</i>
Revision(s) Affected	0, A
Details	<p>When a Frame Error (FE) occurs during checksum byte reception, then the next incoming header is treated as data and another FE will be flagged. The incoming header will be received correctly and the communication happens correctly. This is applicable for both Master and Slave nodes.</p>
Workaround(s)	<p>Criticality and software workaround applicability depend on the application response to the FE flag. Following are the possible scenarios:</p> <p>(a) FE occurred during checksum byte reception followed by a No Response Error (NRE). Comment: If the application issues a software reset in the No Response Error Interrupt Service Routine (NRE ISR), it may result in the incoming header being lost. The application has to handle this situation in software.</p> <p>(b) FE occurred during checksum byte reception followed by an NRE; and if the next header comes before the assigned slot (i.e., before servicing FE and NRE), then another FE will occur. Comment: Application has to ignore the second FE.</p> <p>There will not be any impact on the incoming header reception or communication.</p>
Advisory	<i>LIN: When LIN Slave Adapts Itself to a Faster Incoming Header, No Response Error (NRE)/Time-out Might not get Flagged Within Expected Time Frame</i>
Revision(s) Affected	0, A
Details	<p>No Response Error (NRE) should get flagged if the response reception is not completed within T_{FRAME_MAX} time frame and time-out should get flagged if there is no bus activity for 4 seconds. However, when a LIN slave adapts itself to a faster incoming header, NRE/time-out might not get flagged in the expected time frame. This issue is applicable to SLAVE mode only and ADAPTIVE mode (ADAPT bit set) only.</p>
Workaround(s)	Program BRSR register to a faster baud rate than the expected bus baud rate value.

Advisory	<i>LIN: LIN Rejects a Valid Byte Followed Upon After a False Start Bit</i>
Revision(s) Affected	0, A
Details	LIN rejects the first byte followed upon after a data byte with a false start bit. When a false start bit is received during Identifier or response data, then the next received byte will be rejected.
Workaround(s)	<p>Following are the scenarios which can occur in this situation:</p> <p>(a) False start bit occurs during the data byte reception. In this case, along with the existing byte, next data byte also will get rejected. A No Response Error (NRE) will occur in this case. Action: Issue a software reset in the No Response Error Interrupt Service Routine (NRE ISR).</p> <p>(b) False start bit occurs during the checksum byte. In this case, the next valid start bit evaluation will happen on reception of a falling edge or on reception of an incoming synch break. The checksum byte will be rejected in this case and the CE flag may get set depending on the expected checksum byte. In either case, the next incoming header will be received correctly and the module re-synchronizes to the incoming header. An NRE will occur in this case. Action: Issue a software reset in the NRE ISR. Note: If the next header is received before servicing the NRE, then a Frame Error (FE) occurs (since the next sync break is treated as a checksum byte). This FE should be ignored by the software. The application has to ensure that all pending LIN interrupts are serviced before the reception of the next header.</p> <p>(c) False start bit occurs during Identifier field. In this case, the incoming data byte will be treated as ID, and an ID parity error (PE) occurs. Action: Issue a software reset in the ID PE Interrupt Service Routine (ISR).</p> <p>In all the above cases, the LIN module never misses an incoming header and always re-synchronizes to the incoming header.</p>

Advisory	<i>LIN: Incomplete Synch Field Reception Results in Loss of Next Header</i>
Revision(s) Affected	0, A
Details	When the node is receiving the header and if the synch field reception is incomplete, then the next header will be missed. The node will flag an Inconsistent Synch Field Error (ISFE) when the next header is received and discard the header. Subsequent headers will be received correctly. So, overall, the node will miss one header after the faulty synch field reception. This issue is applicable for both MASTER and SLAVE mode (since the master node will receive the value on the bus during header transmission).
Workaround(s)	<p>Master mode: Do a software reset before triggering the next header.</p> <p>Slave mode: None.</p>

Advisory	<i>LIN: When a Slave Receives a Wake-up Request With Length > 500 μs, it Does Not Transmit a Response to the First Header Following the Wake-up</i>
Revision(s) Affected	0, A
Details	After waking up, the slave either does not transmit a response or transmits a corrupted response.
Workaround(s)	None

Advisory — *LIN: Incomplete Frame Header Consisting of Only Sync Break Will Cause the Following Complete Header to be not Received by Slave* www.ti.com

Advisory *LIN: Incomplete Frame Header Consisting of Only Sync Break Will Cause the Following Complete Header to be not Received by Slave*

Revision(s) Affected 0, A

Details If the slave receives a header that is only a Sync Break (with no Sync Field), and the next incoming header is a complete header, then the slave gets stuck and is not able to receive the latest incoming header (and the receive buffer does not show the latest incoming header's ID). If the incomplete header consists of a Sync Break + Sync Field, then the next incoming header can be received with no error. The issue is when the incomplete header is a Sync Break only. A Sync Break alone may cause a Bit Error followed by Frame Error (as the node is receiving as well). And when the next ID is transmitted, an Inconsistent Synch Field Error (ISFE) will be seen and that ID ignored.

Workaround(s) **Master mode:** Do a software reset before triggering the next header.

Slave mode: None.

Advisory***Boot-ROM: eCAN Boot Mode Not Available on TMX Samples***

Revision(s) Affected

0

Details

eCAN boot mode is not available in TMX samples. If OTP_KEY and OTP_BMODE are programmed with 0x55AA and 0x0007 respectively (boot from eCAN), the device will instead boot to flash.

Workaround(s)

None. This will be fixed in TMS silicon.

Advisory***Boot ROM: Flash API [Flash_Erase() Function]***

Revision(s) Affected

0, A

Details

The Flash API (V1.00) in the device boot ROM contains a coding error in the Flash_Erase() function. This issue does not affect the flash erase operation. On an Erase failure, the function would fail and return error status as expected, but the error address field does not get updated correctly.

Workaround(s)

None

Advisory	<i>Zero-Pin Oscillator: Modification to Oscillator Frequency Parameter</i>
Revision(s) Affected	0, A
Details	The zero-pin oscillator is now specified with the center frequency at a defined temperature and temperature coefficient to calculate the absolute frequency at any operational temperature. Customers will need to check their temperature profile to ensure the zero-pin oscillator meets their requirement.
Workaround(s)	If the frequency output does not meet a needed tolerance, software compensation can be used to achieve better than 1% frequency spread about 10 MHz at any temperature.

Advisory *eQEP: Missed First Index Event*

Revision(s) Affected 0, A

Details

If the first index event edge at the QEPI input occurs at any time from one system clock cycle before the corresponding QEPA/QEPB edge to two system clock cycles after the corresponding QEPA/QEP edge, then the eQEP module may miss this index event. This can result in the following behavior:

- QPOSCNT will not be reset on the first index event if QEPCTL[PCRM] = 00b or 10b (position counter reset on an index event or position counter reset on the first index event).
- The first index event marker flag (QEPSTS[FIMF]) will not be set.

Workaround(s)

Reliable operation is achieved by delaying the index signal such that the QEPI event edge occurs at least two system clock cycles after the corresponding QEPA/QEPB signal edge. For cases where the encoder may impart a negative delay (t_d) to the QEPI signal with respect to the corresponding QEPA/QEPB signal (i.e., QEPI edge occurs before the corresponding QEPA/QEPB edge), the QEPI signal should be delayed by an amount greater than " $t_d + 2 \cdot \text{SYSCLKOUT}$ ".

Advisory *eQEP: eQEP Inputs in GPIO Asynchronous Mode*

Revision(s) Affected 0, A

Details

If any of the eQEP input pins are configured for GPIO asynchronous input mode via the GPxQSELn registers, the eQEP module may not operate properly. For example, QPOSCNT may not reset or latch properly, and pulses on the input pins may be missed. This is because the eQEP peripheral assumes the presence of external synchronization to SYSCLKOUT on inputs to the module.

For proper operation of the eQEP module, input GPIO pins should be configured via the GPxQSELn registers for synchronous input mode (with or without qualification). This is the default state of the GPxQSEL registers at reset. All existing eQEP peripheral examples supplied by TI also configure the GPIO inputs for synchronous input mode.

The asynchronous mode should not be used for eQEP module input pins.

Workaround(s)

Configure GPIO inputs configured as eQEP pins for non-asynchronous mode (any GPxQSELn register option except "11b = Asynchronous").

Advisory	<i>Watchdog: Incorrect Operation of CPU Watchdog When WDCLK Source is OSCCLKSRC2</i>
Revision(s) Affected	0, A
Details	When OSCCLKSRC2 is used as the clock source for CPU watchdog, the watchdog may fail to generate a device reset intermittently.
Workaround(s)	WDCLK should be sourced only from OSCCLKSRC1 (INTOSC1). The CPU may be sourced from OSCCLKSRC2 or OSCCLKSRC1 (INTOSC1).

Advisory	<i>Oscillator: CPU Clock Switching to INTOSC2 May Result in Missing Clock Condition After Reset</i>
Revision(s) Affected	0, A
Details	<p>After at least two system resets (not including power-on reset), when the application code attempts to switch the CPU clock source to internal oscillator 2, a missing clock condition will occur, and the clock switching will fail under the following conditions:</p> <ul style="list-style-type: none"> • X1 and X2 are unused (X1 is always tied low when unused). • GPIO38 (muxed with TCK and XCLKIN) is used as JTAG TCK pin only. • JTAG emulator is disconnected. <p>The missing clock condition will recover only after a power-on reset when the failure condition occurs.</p>
Workaround(s)	<p>Before switching the CPU clock source to INTOSC2 via the OSCCLKSRCSEL and OSCCLKSRC2SEL bits in the CLKCTL register, the user must toggle the XCLKINOFF and XTALOSCOFF bits in the CLKCTL register as illustrated in the below sequence:</p>

```

CLKCTL |= 0x6000;      // XCLKINOFF = 1, XTALOSCOFF = 1
CLKCTL &=~0x6000;     // XCLKINOFF = 0, XTALOSCOFF = 0
CLKCTL |= 0x6000;     // XCLKINOFF = 1, XTALOSCOFF = 1
CLKCTL &=~0x6000;     // XCLKINOFF = 0, XTALOSCOFF = 0
CLKCTL |= 0x6000;     // XCLKINOFF = 1, XTALOSCOFF = 1

```

Once the above procedure is executed, then the OSC2 selection switches can be configured.

If the JTAG emulator is connected, and GPIO38 (TCK) is toggling, then the above procedure is unnecessary, but will do no harm.

If no clock is applied to GPIO38, it is also recommended that a strong pullup resistor on GPIO38 be added to V_{DDIO} .

5 Documentation Support

For device-specific data sheets and related documentation, visit the TI web site at: <http://www.ti.com>.

For further information regarding the Piccolo devices, see the *TMS320F28030, TMS320F28031, TMS320F28032, TMS320F28033, TMS320F28034, TMS320F28035 Piccolo Microcontrollers Data Manual* (literature number [SPRS584](#)).

6 I-to-J Revision History

This revision history highlights the technical changes made to the SPRZ295I errata document to make it an SPRZ295J revision.

Scope: See table below.

LOCATION	ADDITIONS, DELETIONS, AND MODIFICATIONS
Section 4	Known Design Marginality/Exceptions to Functional Specifications: <ul style="list-style-type: none"> • Added "ADC: Offset Self-Recalibration Requirement" advisory

7 H-to-I Revision History

This revision history highlights the technical changes made to the SPRZ295H errata document to make it an SPRZ295I revision.

Scope: See table below.

LOCATION	ADDITIONS, DELETIONS, AND MODIFICATIONS
Section 4	Known Design Marginality/Exceptions to Functional Specifications: <ul style="list-style-type: none"> • Added "eQEP: eQEP Inputs in GPIO Asynchronous Mode" advisory

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products

Audio	www.ti.com/audio
Amplifiers	amplifier.ti.com
Data Converters	dataconverter.ti.com
DLP® Products	www.dlp.com
DSP	dsp.ti.com
Clocks and Timers	www.ti.com/clocks
Interface	interface.ti.com
Logic	logic.ti.com
Power Mgmt	power.ti.com
Microcontrollers	microcontroller.ti.com
RFID	www.ti-rfid.com
OMAP Mobile Processors	www.ti.com/omap
Wireless Connectivity	www.ti.com/wirelessconnectivity

Applications

Automotive and Transportation	www.ti.com/automotive
Communications and Telecom	www.ti.com/communications
Computers and Peripherals	www.ti.com/computers
Consumer Electronics	www.ti.com/consumer-apps
Energy and Lighting	www.ti.com/energy
Industrial	www.ti.com/industrial
Medical	www.ti.com/medical
Security	www.ti.com/security
Space, Avionics and Defense	www.ti.com/space-avionics-defense
Video and Imaging	www.ti.com/video

TI E2E Community Home Page

e2e.ti.com

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2012, Texas Instruments Incorporated